Journal of Symbolic Computation 46 (2011) 219–221

Contents lists available at ScienceDirect

Journal of Symbolic Computation

journal homepage: www.elsevier.com/locate/jsc

Foreword

This special issue consists of articles selected from a collection of papers that have been invited following our workshops on Applied and Computational Category Theory (ACCAT) organized as satellite events at the European Joint Conferences on Theory and Practice of Software, ETAPS 2008 and ETAPS 2009.

The originators of the ACCAT workshops are Hartmut Ehrig (Berlin) and Jochen Pfalzgraf (Salzburg); they started with the 1st ACCAT workshop at ETAPS 2006.

Category theory is a well-known powerful mathematical modeling language with a wide area of applications in mathematics and computer science, including especially the semantical foundations of topics in software science and development. For about 30 years there have been workshops co-organized by H. Ehrig including these topics. More recently, the ACCAT group established by J. Pfalzgraf at Linz and Salzburg has begun to study interesting applications of category theory in geometry, neurobiology, cognitive sciences, and artificial intelligence. It is the intention of the ACCAT workshops to bring together leading researchers in these areas with those in software science and development in order to transfer categorical concepts and theories in both directions.

The organizers are representatives as regards categorical methods for several areas, like geometry, neurobiology, cognitive sciences, and artificial intelligence on one hand and software science and development on the other hand. Categorical methods are already well-established for the semantical foundation of type theory (Cartesian closed categories), automata theory (monoidal (pseudo-)closed categories), data type specification frameworks (institutions) and graph transformations (adhesive high-level replacement categories), which are most relevant for ETAPS.

Saunders Mac Lane, one of the founders of category theory (CAT), writes in the introduction of his famous book *Categories for the Working Mathematician, Springer-Verlag, Graduate Texts in Mathematics, Vol. 5*: "Category theory starts with the observation that many properties of mathematical systems can be unified and simplified by a presentation with diagrams of arrows".

In the *Handbook of Logic in Computer Science, Vol. 1*, Axel Poigné makes the following comments: "If asked for a single reason for the attention that category theory, at least as a language, enjoys in some areas of computer science, I would guess that its attraction stems from being a foundational theory of functions which provides a sound basis for (functional) programming and programming logic. If asked for more reasons I would recollect the familiar arguments namely that category theory formalizes otherwise vague concepts, provides a language that brings to the surface common basic concepts in ostensibly unrelated areas, allows us to translate problems from one area to another where a solution may be more easily achieved, or more specifically with regard to computer science, category theory allows easier access to various areas of mathematics in that it provides a core of properties to be looked for, offers a rich language in which to axiomatize, differentiate and compare structures in computer science and mathematics".

This special issue contains the following five papers that were submitted by invitation and are based on the authors' contributions to the ACCAT workshops 2008 and 2009.

0747-7171/$ – see front matter
doi:10.1016/j.jsc.2010.09.005

A lattice-theoretical perspective on adhesive categories

Paolo Baldan, Filippo Bonchi, Andrea Corradini, Tobias Heindel and Barbara König

In an adhesive category, subobjects of an object form a distributive lattice. Baldan et al. therefore combine lattice theory with the theory of adhesive categories to obtain conclusions about subobjects. The paper establishes several results relating objects in adhesive categories to structures corresponding to graphs. It is shown that any finite object of an adhesive category can be generated as a colimit of its irreducible subobjects, where the colimit is a so-called Van Kampen colimit. In addition, the relationship between arrows in an adhesive category and homomorphisms between subobject lattices is analyzed and characterized.

How to delete categorically --- two pushout complement constructions

Benjamin Braatz, Ulrike Prange and Thomas Soboll

In category theory, no direct construction for deletion exists as is given, for example, in set theory by the complement. Braatz et al. analyze this problem, which is important as a main operation in rule-based transformation systems. While in the DPO approach pushout complements are used, these do not exist or are not unique in general categories. Two different deletion constructions – based on initial pushouts and quasi-coproduct complements – are suggested and shown to be applicable in the relevant categories.

Cartesian effect categories are Freyd-categories

Jean-Guillaume Dumas, Dominique Duval and Jean-Claude Reynaud

The authors present a new categorical approach to be used as the basis for the definition of the semantics of programming languages, in terms of the new notion of Cartesian effect categories. A basic idea of the approach is to consider that the morphisms in a Cartesian effect category K include a special class of "pure" morphisms or, equivalently, that a Cartesian effect category comes with a wide subcategory C consisting of all pure morphisms. The intuition behind pure morphisms is that they are the "pure functions" (without side-effects, or non-termination, or that they cause no errors). Furthermore, it is assumed that C has a final object. This object is used to define an equivalence relation between morphisms that describes whether two functions have a pure behavior on the same subdomain.

Symbolic graphs for attributed graph constraints

Fernando Orejas

Constraints are an important concept for modeling and specification, and for restricting the applicability of transformation rules. Orejas presents a new approach to attributed graphs and graph transformation using symbolic graphs. Within these, attribute values are not represented by edges to special data nodes, but constrained by a set of formulas over attribute variables. A new notion of attributed graph constraints allows us to formulate a set of inference rules obtaining a sound and complete proof system for reasoning about typed attributed graphs.

A categorical framework for the transformation of object-oriented systems: Models and data

Christoph Schulz, Michael Löwe and Harald König

Rewriting class and object diagrams is a non-trivial task mainly due to the fact that the inheritance relations do not integrate well with the notion of homomorphisms. Schulz et al. present a well-defined categorical framework for specifying refactorings in object-oriented systems. Basically, systems can be described as typed graphs with some additional constraints. A refactoring is then described by a schema transformation, which is a span of schemas. The construction is based on a migration functor, which is the composition of a pullback functor, a composition functor, and an epireflector. It is shown that refactoring is correct in the sense that migration respects the typing axioms. The theory is explained with several refactoring examples, where a model of an insurance company is transformed.

Applied and Computational Category Theory, ACCAT for short, was the name of one of J. Pfalzgraf's working groups which he established during his time (1990–1996) in the faculty at RISC-Linz, the Research Institute for Symbolic Computation, Johannes Kepler University, Linz, Austria. At that time Hoon Hong, the editor in chief of JSC, worked in the neighboring office — they had many interesting discussions about various topics, including CAT and applications. We are grateful to Hoon Hong for inviting us to be guest editors for a special issue of JSC, presenting publications dealing with ACCAT topics. Our thanks go to the authors of this special issue and all our referees for their valuable work.

Jochen Pfalzgraf
Universität Salzburg,
Fachbereich Computerwissenschaften,
Jakob Haringer Str. 2,
5020 Salzburg, Austria
E-mail address: jpfalz@cosy.sbg.ac.at.

Hartmut Ehrig
Ulrike Golas
Technische Universität Berlin,
Institut für Softwaretechnik und Theoretische Informatik,
Franklinstr. 28/29,
D-10587 Berlin, Germany
E-mail addresses: ehrig@cs.tu-berlin.de (H. Ehrig), ugolas@cs.tu-berlin.de (U. Golas).

Thomas Soboll
Universität Salzburg,
Fachbereich Computerwissenschaften,
Jakob Haringer Str. 2,
5020 Salzburg, Austria
E-mail address: tsoboll@cosy.sbg.ac.at.

15 September 2010
Available online 25 September 2010